

# Manual of



## Authors

Lázaro Guillermo Pérez Montoto  
Francisco Javier Prado Prado  
Cristian R. Munteanu  
Humberto González Díaz

2009

([lgpm2002@yahoo.es](mailto:lgpm2002@yahoo.es))

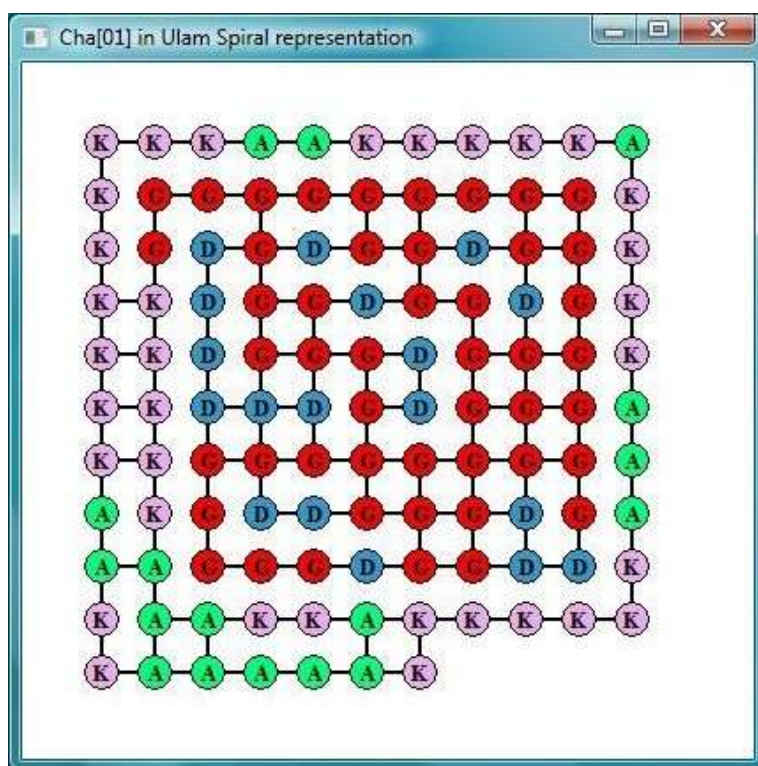


## What is CULSPIN?

CULSPIN transforms any sequence of letters in a graphic representation that uses as template the spiral of Ulam (disposition of the natural numbers in spiral form) and connects those nodes that belongs to the same class (they have the same letter). For example:

Cha [01]

GDDGGDGGGGGGGDDGGGDGDDGGGDGGGDGDDGGDDGGGGGGGGGGGGGGGGGGKKKKKAAAKKAKKKKKKAAA  
KKKKAKKKKKAANKKKKKKKKAANKAAAAA



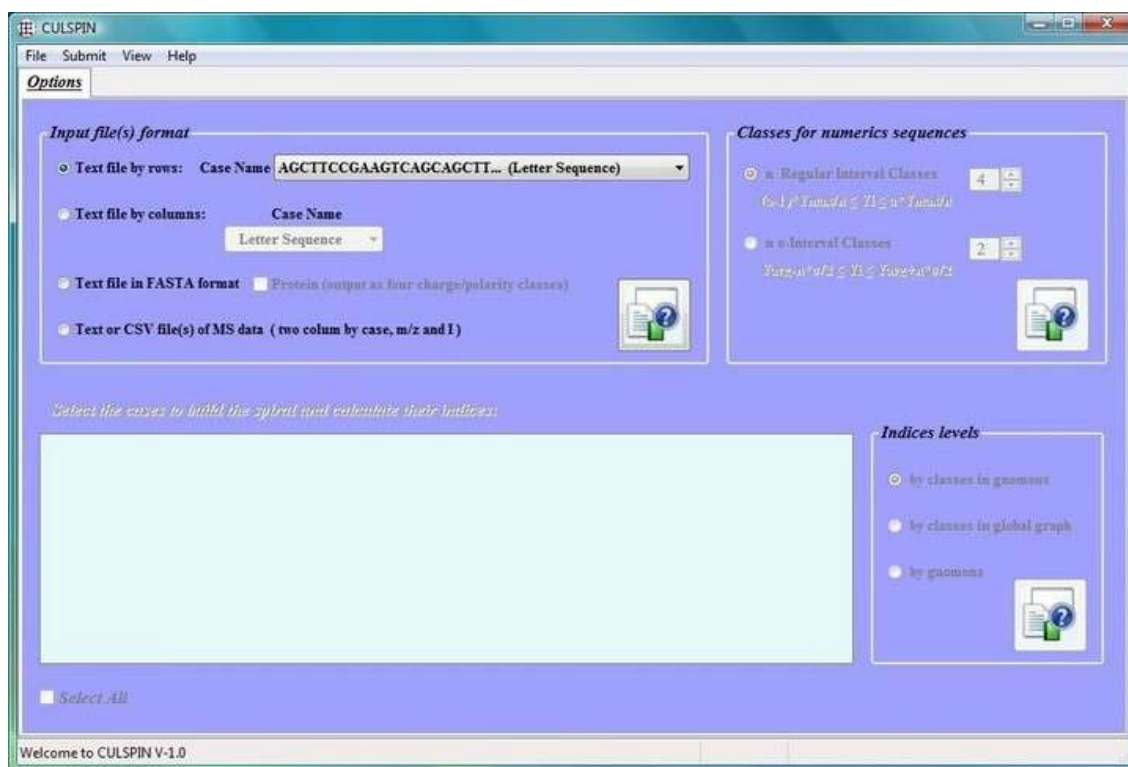
Also, using this graph, CULSPIN calculates two families of Topological Indices (TIs). These indices can be calculated at several levels: for each one of the classes in each Ulam gnomon, for each one of the classes in the whole graph and for each gnomon independent of the class type. On the other hand, the 2D graph (U-graphs) generated by the application, besides being able to be visualized, they can be exported in order to use them in other external programs to calculate another families of TIs. All the numeric indices can be saved and/or exported to submit them later on to a great variety of statistical analysis or to create QSAR models (quantitative structure-activity relationship). Examples of sequences are the amino acids chains in proteins, the nucleic acids and the mass spectra of proteins. CULSPIN can be used to study different systems, from the simple systems of atoms in anti-tumor small molecules, until complex systems of metabolic, social, computational or biological systems nets.

## ¿What CULSPIN can do?

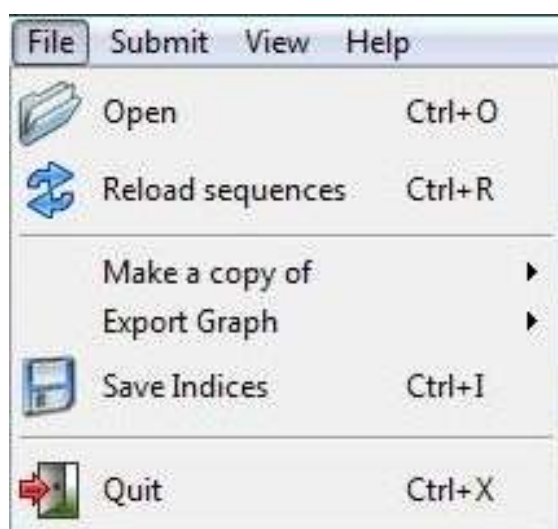
- **Read** sequences of letters organized by rows or columns starting from **TXT** files;
- **Read** sequences represented in **FASTA** format starting from **TXT** files;
- **Read** numeric sequences or series, organized by rows or columns starting from **TXT** files;
- **Read** Mass Spectra (**MS**) data starting from multiple **TXT** or **CSV** files;
- **Transform** numeric sequences or series and **MS** data in letters sequences;
- **Transform** any letters sequence in their corresponding U-graph connecting the nodes that belong to the same class (they have the same letter);
- **Compute** two families of **TIs** using the generated **U-graphs** and **Show** their values in a grid page;
- **Plot** and **Display** the **U-graph** of the selected sequence;
- **Export** the connectivity information of each one of the **U-graphs** in a **CT** or **NET** file;
- **Save** the calculated **TIs** in **TXT** or **CSV** file.

## How to use CULSPIN?

CULSPIN is an interactive application created with Python/wxPython with a notebook format and that presents a main menu bar with the following options:



### File :



- **Open file** : it allows to select the input file(s) for open it and to upload the data. Once finished the reading, the sequences are shown in a list box.

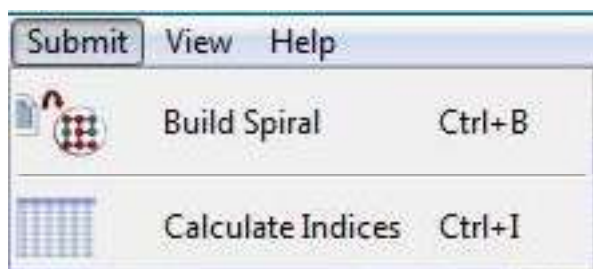
- **Reload sequences** : it allows working again with all the original sequences. It is only activate when we was not built the spiral to some original sequence. Once finished, all the initial sequences are shown in a list box.

- **Make a copy of** : save in a *.txt* file the original sequences or those that

were studied, organizing them such and like that they are shown in the list box. It is activate when the input data did not have this format.

- **Export graph** : to export the connectivity information of alls *U-graphs* in **.ct** or **.net** files in order to open them with other programs to make another calculations.
- **Save Indices** : to store the indices values in one **.txt** or **.csv** file for the posterior statistical analysis.
- **Quit** : to exit of application.

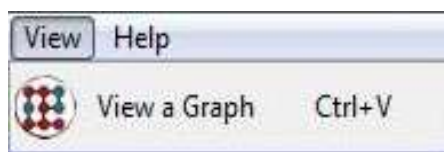
### Submit :



- **Build Spiral** : to Build the *U-graph* and connect the nodes with the same class to each one of the selected sequences .
- **Calculate Indices** : to compute the *TIs* using the *U-graph* of the selected sequences. Once finished

the operation the results are available in a new page of the notebook.

### View :



- **View a graph** : to plot and to visualize, in an independent window, the *U-graph* of one selected sequence. It is only enable after a **Build Spiral** operation.

### Help :



- **Help** : to show the help.
- **About** : to show the classic about windows.

```
Cha[01] Cha[02] Cha[03]
D G G
D D D
G D G
G G G
D G D
G D G
G G G
G G G
G G G
G G G
G G G
G G G
```



**Numeric sequences:**

```

Cha[01] Cha[02] Cha[03]
-7.86E-05 2.18E-07 9.60E-05
2.18E-07 9.60E-05 0.00036601
9.60E-05 0.00036601 0.0008102
0.00036601 0.0008102 0.00142856
0.0008102 0.00142856 0.00222112
0.00142856 0.00222112 0.00318787
0.00222112 0.00318787 0.00432881
0.00318787 0.00432881 0.00564393
0.00432881 0.00564393 0.00713324
0.00564393 0.00713324 0.00879674
0.00713324 0.00879674 0.01063443
0.00879674 0.01063443 0.01264631
0.01063443 0.01264631 0.01483238
0.01264631 0.01483238 0.01719263
0.01483238 0.01719263 0.01972708

```

**c-) Text file in FASTA format:**

```

>gi|221068402|ref|ZP_03544507.1|enzyme [Comamonas testosteroni KF-1]
MSEPVNQWPQTLEERIDRLSLDAIRQLAGKYSLSLDMRMDAHVNLFPDIAKVGKEKVGRAHFMWQDS
TLRDQFTGTSHHLGQHIIEFVDRDHATGVVYSKNEHECGAEWVIMQMLYWDDYERIDGQWYFRRRLPCYW
YATDLNKKPPIGDMKMRWPGREPYPHGAFFELFSPWKEFWAQRPGKDQLPQVAAPAPLEQFLRTMRRGTPAP
RMRVR

>gi|220713425|gb|EED68793.1| enzyme [Comamonas testosteroni KF-1]
MSEPVNQWPQTLEERIDRLSLDAIRQLAGKYSLSLDMRMDAHVNLFPDIAKVGKEKVGRAHFMWQDS
TLRDQFTGTSHHLGQHIIEFVDRDHATGVVYSKNEHECGAEWVIMQMLYWDDYERIDGQWYFRRRLPCYW
YATDLNKKPPIGDMKMRWPGREPYPHGAFFELFSPWKEFWAQRPGKDQLPQVAAPAPLEQFLRTMRRGTPAP
RMRVR

>gi|77360245|ref|YP_339820.1| enzyme [Pseudoalteromonas haloplanktis TAC125]
MQYLVISDIYGKTPCLQQLAKHFNAENQIVDPYNGVHQALENEEEYKLFIKHCGHDEYAAKLEEFNKL
SKPTICIAFSAGASAAWRAQASTTTTHLKKVIAFYPTQIRNYLNIDAIHPCEFIFFGFEPHFNVDELITN
LSAKNNVRCLKTLYLHGFMNQSQNFSEYGYQYFYKVIKTANSEAH

```

**Note:** In the cases of proteins, if the option **Protein** is selected, each amino acid present in the sequences is codified in one of the four different amino acids classes determined by their side chains properties: non-polar and neutral; polar and neutral; acidic and polar; and basic and polar.

**d-) Text or CSV files of MS data:** In this option each case is stored in an independent file. In them the **MS** signal data of are organized in two columns: **mass/charge (m/z)** and **Intensity** with header or not. The files can be in **TXT** o **CSV** format.

**TXT files:** (the columns are separated by tabulation)

```

2.5660      0.6601
3.6601      8.9102
8.1024      42.0856
14.2856     22.2112
22.2112     3.8787
31.8787     4.3288
43.2881     56.4393
56.4393     71.3324
71.3324     87.9674
87.9674     90.0000
106.3443    12.1631
126.4631    8.3238
148.3238    100.9263

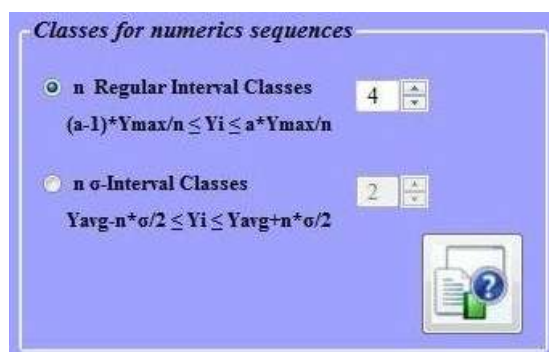
```



**CSV files:** (the elements are separated by commas)

```
m/z, Intensity
2.5660, 0.6601
3.6601, 8.9102
8.1024, 42.0856
14.2856, 22.2112
22.2112, 3.8787
31.8787, 4.3288
43.2881, 56.4393
56.4393, 71.3324
71.3324, 87.9674
87.9674, 90.0000
106.3443, 12.1631
126.4631, 8.3238
148.3238, 100.9263
```

**II- Classes for numerics sequences:** this spin control box will be enable when a numeric input format is selected and it offers two different heuristics to transform the numeric data into letters sequences.



- ***n Regular Interval Classes:*** in this option numeric data is divided in n intervals or classes ( $2 \leq n \leq 10$ ) and to each one of them is assigned a letter. This way, the elements or signs of the numeric sequence are coded with the letter from the class to which belongs.
- ***n σ-Interval Classes:*** in this option the numeric data is divided in  $2n+2$  intervals ( $2 \leq n \leq 4$ ) that are function of its standard deviation. To each one of class is assigned a letter and the element or signal of the numeric sequence are code with the letter from the class to which belongs.

**Note:** In the cases of MS data, this program version transforms the original data in numeric sequences obtained by means of the m/z and intensity values multiplication and after that, it makes the transformation in the letters sequences using the heuristic selected by de user.

**III- A list box for view/select sequences:** this list box has the function to show and to allow the selection of letters sequences or cases. At the beginning, the list is empty but after reading the data starting from the input file, the list shows the directly read letters sequences or those that was obtained by means of some previous encode or

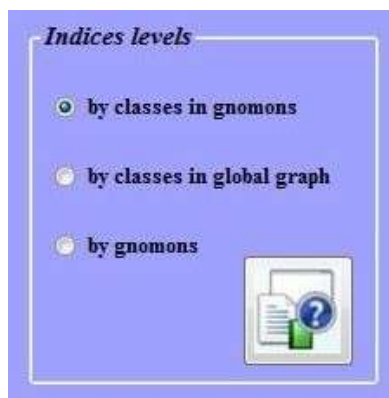
[illegible]

Select the cases to calculate indices or one case to view its spiral:

Cha[01]	GDDGGDGGGGGGGDDGGGDGDDGGGDGGGDGDDDDGGGGGGDGDDGGGGGGGGGGGGGGKKKKKAAAKKA
Cha[02]	DDGGDGGGGGGGDDGGGDGDDGGGDGGGDGDDDDGGGGGGDGDDGGGGGGGGGGGGGGKKKKKAAAKKA
Cha[05]	GDDGDGGGGGGGDDGGGDGDDGGGDGGGDGDDDDGGGGGGDGDDGGGGGGGGGGGGGGKKKKKAAAKKA
Cha[06]	GDDGDGGGGGGGDDGGGDGDDGGGDGGGDGDDDDGGGGGGDGDDGGGGGGGGGGGGGGKKKKKAAAKKA
Cha[09]	GDDGGDGGGGGGGDDGGGDGDDGGGDGGGDGDDDDGGGGGGDGDDGGGGGGGGGGGGGGKKKKKAAAKKA

☐ Select All

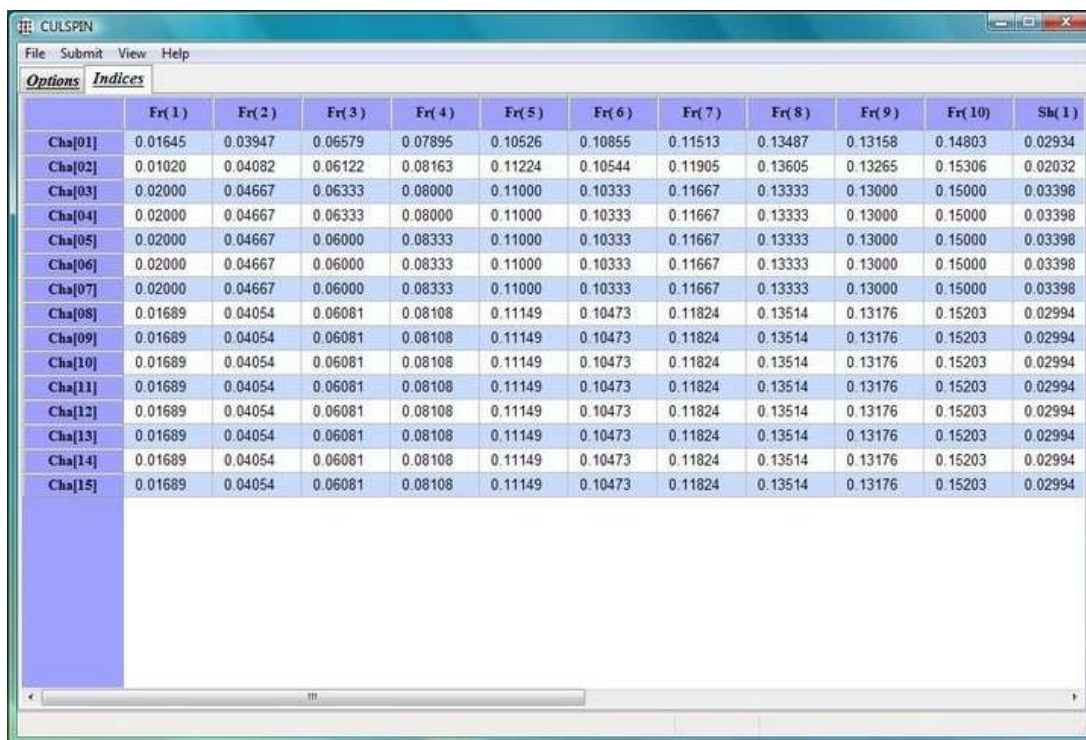
**IV- Indices levels:** this controls box is only activate if it has been built one spiral at least and it allows to select among the three levels implemented in this version of CULSPIN for the ***TIs*** calculation.



- **by classes in gnomons:** if this option is selected, the two families of *TIs* are calculated for each one of the classes in each one of the gnomons. In the case in that a class is not present in a certain gnomon, its **Frequency** and its **Shannon Entropy** in this gnomon are zero. This option is more useful when the sequences have a few classes and they are not very big, otherwise, a too high number of indices would be obtained and therefore its very annoying its further statistical process.
- **by classes in global graph:** in this option the *TIs* are calculated for each one of the classes but in the whole graph. In other words, the *TIs* of a given class in the whole graph, are the sum of their values in all the gnomons. This option reduces the number of *TIs* in the case of very big sequences for what is a good option in such cases.
- **by gnomons:** if this option is selected, the *TIs* are calculated at gnomons level and independently of the classes. In other words, the indices for a certain gnomon are the sum of the *TIs* of all the classes in this gnomon. This option can be very useful if the sequences have great number of classes and a moderate size.

## **Indices page:**

This page is added to the notebook and it is shown to the user immediately after the *TIs* of the selected sequences are calculated. The page has a grid format in which the columns label are the *TIs* names and the rows labels are the names of the cases or letters sequences.



	Fr( 1 )	Fr( 2 )	Fr( 3 )	Fr( 4 )	Fr( 5 )	Fr( 6 )	Fr( 7 )	Fr( 8 )	Fr( 9 )	Fr( 10 )	Sh( 1 )
Cha[01]	0.01645	0.03947	0.06579	0.07895	0.10526	0.10855	0.11513	0.13487	0.13158	0.14803	0.02934
Cha[02]	0.01020	0.04082	0.06122	0.08163	0.11224	0.10544	0.11905	0.13605	0.13265	0.15306	0.02032
Cha[03]	0.02000	0.04667	0.06333	0.08000	0.11000	0.10333	0.11667	0.13333	0.13000	0.15000	0.03398
Cha[04]	0.02000	0.04667	0.06333	0.08000	0.11000	0.10333	0.11667	0.13333	0.13000	0.15000	0.03398
Cha[05]	0.02000	0.04667	0.06000	0.08333	0.11000	0.10333	0.11667	0.13333	0.13000	0.15000	0.03398
Cha[06]	0.02000	0.04667	0.06000	0.08333	0.11000	0.10333	0.11667	0.13333	0.13000	0.15000	0.03398
Cha[07]	0.02000	0.04667	0.06000	0.08333	0.11000	0.10333	0.11667	0.13333	0.13000	0.15000	0.03398
Cha[08]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[09]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[10]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[11]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[12]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[13]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[14]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994
Cha[15]	0.01689	0.04054	0.06081	0.08108	0.11149	0.10473	0.11824	0.13514	0.13176	0.15203	0.02994

In this grid you can select one cell; a range of cells; or all the cells and copy the selection into clipboard using **Ctrl+C** combination. It is a rapid export procedure that allows to paste these values in another external application such as Excel.

## *Ulam spiral*

In 1963 the mathematical Stanisław M. Ulam discovered certain interesting aspects related with the disposition that adopt the prime numbers when placing the natural numbers in form of a spiral. Then this disposition became highly popularized as visual picture in a number of Scientific American in 1964.

To construct the spiral one must write down a regular grid of numbers, starting with one at the center, and spiraling out the rest of integer numbers just as you can see in the image of below:

101	—	100	—	99	—	98	—	97	—	96	—	95	—	94	—	93	—	92	—	91
102		65	—	64	—	63	—	62	—	61	—	60	—	59	—	58	—	57		90
103		66		37	—	36	—	35	—	34	—	33	—	32	—	31		56		89
104		67		38		17	—	16	—	15	—	14	—	13		30		55		88
105		68		39		18		5	—	4	—	3		12		29		54		87
106		69		40		19		6		1	—	2		11		28		53		86
107		70		41		20		7	—	8	—	9	—	10		27		52		85
108		71		42		21	—	22	—	23	—	24	—	25	—	26		51		84
109		72		43	—	44	—	45	—	46	—	47	—	48	—	49	—	50		83
110		73	—	74	—	75	—	76	—	77	—	78	—	79	—	80	—	81	—	82
111	—	112	—	113	—	114	—	115	—	116	—	117								

In mathematics, this is a simple method of graphing numbers that reveals very hidden patterns in numeric series and sequences. In molecular sciences this Spiral representation was associated to a graph in order to represent DNA nucleotide sequences in a letters sequence of four classes (A, T, G, and C).

## *What is a gnomon?*

The Ulam spiral can be divided in different regions or intervals called gnomons or angular dispositions as one can observe in the following figure:



101	100	99	98	97	96	95	94	93	92	91
102	65	64	63	62	61	60	59	58	57	90
103	66	37	36	35	34	33	32	31	56	89
104	67	38	17	16	15	14	13	30	55	88
105	68	39	18	5	4	3	12	29	54	87
106	69	40	19	6	1	2	11	28	53	86
107	70	41	20	7	8	9	10	27	52	85
108	71	42	21	22	23	24	25	26	51	84
109	72	43	44	45	46	47	48	49	50	83
110	73	74	75	76	77	78	79	80	81	82
111	112	113	114	115	116	117				

To define a gnomon it is necessary to remember the oblong numbers that are those that can be represented by means of the product  $n(n+1)$  with natural  $n$ , that is to say: 2, 6, 12, 20, 30, 42, 56, 72, 90,... These numbers divide to the natural numbers in different intervals growing in size ( $2n$ ). It is easy to see that a serial couple of oblong numbers defines a gnomon and that these angular dispositions leave inserting giving place to rectangles of growing size. It is also clear that each element of the spiral belongs to an only gnomon, thus can be defined, for each element the Ulam coordinate  $U_n$  to the order number of the gnomon to which belongs.

When a sequence of letters is represented in its U-graph, each node is an element of the sequence whose letter represents the class at which this element belongs and in each gnomon one or more different classes will exist.

K	K	K	A	A	K	K	K	K	K	A
K	G	G	G	G	G	G	G	G	G	K
K	G	D	G	D	G	G	D	G	G	K
K	K	D	G	G	D	G	G	D	G	K
K	K	K	A	A	K	K	K	K	K	A
K	G	G	G	G	G	G	G	G	G	K
K	G	D	G	D	G	G	D	G	G	K
K	K	D	G	G	D	G	G	D	G	K
A	A	G	G	G	D	G	G	D	D	K
K	A	A	K	K	A	K	K	K	K	K
K	A	A	A	A	A	K				

## *Indices, definition y calculation*

As it has been commented from the beginning, in the U-graph built using CULSPIN, each node belongs to a certain class and the nodes are not only connected following the sequence of letters, but rather also those nodes that belong to the same class (they have same letter) are connected. So, in our U-graph each node will be connected with one or more nodes. By definition, it is known as node degrees to the number of nodes with those the node in question it is connected and for total degrees of a graph to the sum of the degrees of all the nodes that conform the graph, then we can define as gnomon degrees to the sum of the degrees of the nodes present in this gnomon.

Keeping in mind all the above-mentioned, the indices calculated by CULSPIN are defined and calculated in the following way:

Indices levels	Frequency	Shannon Entropy
by classes in gnomon	$f(c, g) = \frac{\sum_{\substack{n \in c \\ n \in g}} \deg(n(c, g))}{\sum_{i \in G_g} \deg(i)}$ <p><math>c</math>: class; <math>g</math>: gnomon; <math>n_{c,g}</math>: node with class <math>c</math> in gnomon <math>g</math></p>	$Sh(c, g) = -f(c, g) \log(f(c, g))$
by classes in global graph	$f(c) = \frac{\sum_{\substack{n \in c \\ n \in G_U}} \deg(n(c))}{\sum_{i \in G_U} \deg(i)}$ <p><math>c</math>: class; <math>n_c</math>: node with class <math>c</math> in <math>G_U</math></p>	$Sh(c) = -f(c) \log(f(c))$
by gnomons	$f(g) = \frac{\sum_{n \in g} \deg(n(g))}{\sum_{i \in G_U} \deg(i)}$ <p><math>g</math>: gnomon; <math>n_g</math>: node in gnomon <math>g</math></p>	$Sh(g) = -f(g) \log(f(g))$